



Acceso a Datos

PRIMER TRIMESTRE - PROYECTO

Antonio Delgado Vietes | 2ºdam | 13/10/2021

Documentación de Docker

¿Qué es Docker?

La idea principal de Docker es crear contenedores ligeros y portables para las aplicaciones de software que puedan ejecutarse en cualquier máquina con Docker instalado sin tener en cuenta el sistema operativo.

Tener en cuenta que Docker no es un software como VirtualBox o VMware, es decir NO es una máquina virtual, NO es un sistema operativo montado sobre otro.

Docker utiliza el kernel del SO host ahorrando así una mayor cantidad de RAM y CPU.

Instalando docker

Docker es relativamente sencillo de instalar, en este caso explicaré en los dos entornos en los cuales los he instalado, Linux y Windows:

- Windows:
 - En Windows instalé docker en cuestión de minutos simplemente me fui a la [página de descarga](#)
 - Ejecuté el instalador y seguí los pasos de este.
 - Y con estos pasos ya tengo docker instalado
- Linux;
 - En linux si ha sido más compleja la instalación debido a que había que realizar varios pasos:
 - Primero actualicé los repositorios con el comando “ **sudo apt-get update**”
 - Después instale los paquetes necesarios, como certificados, curl, https, etc mediante el comando “**sudo apt-get install apt-transport-https ca-certificates curl gnupg-agent software-properties-common**”
 - A continuación instalé la key que me da docker para ubuntu mediante “**curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add**”
 - Seguidamente agregué el repositorio de docker usando: “ **sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu \$(lsb_release -cs) stable"**”
 - De nuevo volví a actualizar los paquetes e instalé docker con el comando: “**sudo apt-get install docker-ce docker-ce-cli containerd.io**”
 - Y con estos pasos ya tengo instalado docker, para comprobar si funciona probé a ejecutar el comando “**docker run hello-world**” y me devolvió lo siguiente confirmando que estaba instalado

```
Hello from Docker!  
This message shows that your installation appears to be working correctly.
```

DockerHub

DockerHub es el repositorio del cual Docker toma las imágenes que luego provisiona a los contenedores que creamos. Para usar DockerHub me he creado una cuenta, con el usuario lander990. Características de docker hub:

- Tiene repositorios para la gestión de las imágenes de los contenedores.
- Dispone de creación de equipos y organizaciones: Para la gestión del acceso a los depósitos privados de imágenes
- Dispone de imágenes oficiales proporcionadas por Docker como imágenes de editores que las proporcionan proveedores externos y que están certificadas.

Imágenes de docker

Una imagen de Docker es en pocas palabras una instantánea de un contenedor, se puede decir que es la plantilla de un contenedor, es lo que le da un estado inicial y podemos usarla tantas veces como queramos. Las imágenes las podemos encontrar principalmente en DockerHub.

Para ver las imágenes que tengo instaladas puedo verlas en windows con el programa “Docker Desktop” y en Linux con el comando `sudo docker images`:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
hello-world	latest	feb5d9fea6a5	2 months ago	13.3kB

Implementando un contenedor en Docker (Windows)

Instalar un contenedor a través de una imagen es sencillo y rápido utilizando docker, estos son los pasos que he usado para instalar un contenedor de mysql:

1. Primero me he traído mediante un pull la imagen de mysql usando el comando **“docker pull mysql”**
2. A continuación he creado un contenedor con la última versión y lo he llamado con la primera inicial de mi nombre seguido de mi apellido. Además también he configurado el usuario root y le he puesto una contraseña de prueba. **Esto es muy muy poco seguro debido a que la contraseña del root (super-usuario) le estamos poniendo la contraseña en texto plano.** He usado el comando: **“docker run --name mysql-adelgado -e MYSQL_ROOT_PASSWORD=prueba -d mysql:latest”**
3. Después he accedido al contenedor con el comando **“docker exec -it mysql-adelgado bash”**

4. Y con esto ya tendría un contenedor con mysql instalado y preparado en cuestión de minutos, podría acceder al servicio mediante el comando “mysql -u root -p” y poniendo la contraseña que en este caso es “prueba”

Ciclo de vida de un contenedor

Si por ejemplo quiero ver los contenedores que tengo implementados uso el comando “**docker container ls**”, lo que al ejecutarlo se vería algo así

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

A continuación si quiero iniciar un contenedor puedo usar el comando “ **docker container run -it mysql-adelgado**”

No hace falta usar el nombre para iniciar un contenedor, también puedo usar la id (Completa o parcial).